IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Applicant	:	John D. Irish et al.
Application No.	:	10/670,703
Filed	:	September 25, 2003
For	:	METHODS AND APPARATUS FOR ALLOCATING MEMORY
Examiner	:	Midys Rojas
Art Unit	:	2185
Confirmation No.	:	6165

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

BRIEF ON APPEAL

Dear Sir:

In response to the decision of the Examiner in the Final Office Action dated October 16, 2008, finally rejecting claims 1-34, a Notice of Appeal was filed on January 16, 2009. Appellants/Applicants submit the following Brief on Appeal in compliance with 37 CFR § 41.37. For the reasons more fully set forth below, it is respectfully submitted that the final rejection of claims 1-34 should be reversed.

REAL PARTY IN INTEREST:

The real party in interest is INTERNATIONAL BUSINESS MACHINES CORPORATION, the assignee of the subject application.

RELATED APPEALS AND INTERFERENCES:

Appellants/Applicants and Appellants/Applicants' legal representative do not know of any prior or pending appeals, interferences or judicial proceedings which may be related to, directly affect or be directly affected by, or have a bearing on, the Board's decision in this appeal.

STATUS OF CLAIMS:

Claims 1-34 have been rejected and are on appeal.

STATUS OF AMENDMENTS:

An Amendment after the Final Office Action (request for reconsideration) was filed on December 1, 2008. The request was considered as indicated by the Advisory Action mailed on December 16, 2008.

SUMMARY OF CLAIMED SUBJECT MATTER:

Claim 1

Independent claim 1 is directed to a method. The method comprises receiving a set of data (e.g., operation 304 of FIG. 3 and as discussed on pages 7-8, lines 15-4 of the specification). The method further comprises determining whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of a memory (e.g., operation 306 of FIG. 3 and as discussed on pages 8-9, lines 5-23). The method further comprises if a free group entry of the size required by the portion of the set of data does not exist in one of the plurality of sections of the memory, determining whether the memory includes one or more sections of an unallocated size (e.g., operation 310 of FIG. 3 and as discussed on pages 9-10, lines 24-13). The method further comprises if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data (e.g., operation 312 of FIG. 3 and as discussed on pages 10-11, lines 14-24).

Claim 18

Independent claim 18 is directed to an apparatus. The apparatus comprises a memory (e.g., memory 106 of FIG. 1 and as discussed on page 5, lines 1-24 of the specification. The apparatus

further comprises a plurality of registers (e.g., registers 116-120 of FIG. 1 and as discussed on page 5, lines 4-24 of the specification). The apparatus further comprises dynamic allocation logic coupled to the memory and the plurality of registers (e.g., dynamic allocation logic 108 of FIG. 1 and as discussed on page 5, lines 12-24 of the specification). The dynamic allocation logic is adapted to receive a set of data (e.g., operation 304 of FIG. 3 and as discussed on pages 7-8, lines 15-4 of the specification). The dynamic allocation logic is further adapted to determine whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of the memory (e.g., operation 306 of FIG. 3 and as discussed on pages 8-9, lines 5-23). The dynamic allocation logic is further adapted to if a free group entry of the size required by the portion of the set of data does not exist in one of the plurality of sections of the memory, determine whether the memory includes one or more sections of an unallocated size (e.g., operation 310 of FIG. 3 and as discussed on pages 9-10, lines 24-13). The dynamic allocation logic is further adapted to if the memory includes one or more sections of an unallocated size, allocate one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data (e.g., operation 312 of FIG. 3 and as discussed on pages 10-11, lines 14-24).

"Means" Or "Step"

None of the claims contain an element expressed as a means or step for performing a specified function without the recital of

structure, material, or acts in support thereof.

GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL:

Claims 1-34 were rejected under 35 U.S.C. § 103(a) as being obvious over the combination of U.S. Patent Publication No. 2001/0011338 to Bonola [hereinafter *Bonola*] and the conventional allocation process discussed in the "Background" section of Applicants' disclosure [hereinafter *conventional memory allocation process*].

ARGUMENT:

REVIEW OF CITED ART

Bonola

Bonola is directed to "a system and method for providing linearly scalable dynamic memory management in a multiprocessing system" *Bonola*, paragraph 2. *Bonola* explains that in multiprocessing systems (e.g., systems including multiple processors for executing multiple threads), previous schemes for managing multiple threads accessing shared memory were slow. *Bonola*, paragraphs 11-12. Accordingly, *Bonola* discusses an alternative approach for managing multiple threads accessing shared memory. *See e.g. Bonola*, paragraph 13.

In paragraphs 37-38, *Bonola* discusses examining a heap subregion "to determine if there is an appropriately sized heap subregion 302A-302F in the free list…" and assigning "the address of the heap subregion 302A-302F contained in the entry 408" to the application. *Bonola* goes on in paragraphs 39-40 to discuss dividing a remaining half of the heap subregion in half if "the remaining half of the heap subregion contains twice as much memory as is required by the requesting application…" This process is "repeated until a heap subregion is yielded which does not have twice as much memory as is required."

The citations to *Bonola* do not discuss, for example, determining whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality sections of memory. Nor do the citations to *Bonola* discuss, for example, if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data.

A PRIMA FACIE CASE OF OBVIOUSNESS OF CLAIMS 1-34 HAS NOT BEEN ESTABLISHED AS IT HAS NOT BEEN SHOWN THAT *BONOLA* DISCLOSES DETERMINING WHETHER A FREE GROUP ENTRY OF A SIZE REQUIRED BY A PORTION OF THE SET OF DATA EXISTS IN MEMORY PRE-ALLOCATED WITH A GROUP SIZE IN ONE OF A PLURALITY OF SECTIONS OF A MEMORY.

Appellants/Applicants respectfully submit that the record fails to establish that each feature of independent claims 1 and 18 is disclosed by the proposed combination of *Bonola* and the *conventional memory allocation process.* Accordingly,

Appellants/Applicants respectfully submit that the record fails to establish a prima facie case of obviousness.

"The key to supporting any rejection under 35 U.S.C. 103 is the clear articulation of the reason(s) why the claimed invention would have been obvious." *MPEP* § 2142 (8th Ed. 2001) (Rev. 7, July 2008). "The Supreme Court in KSR International Co. v. Teleflex Inc., 550 U.S. ___, ___, 82 USPQ2d 1385, 1396 (2007) noted that the analysis supporting a rejection under 35 U.S.C. 103 should be made explicit." *Id.* "The Federal Circuit has stated that 'rejections on obviousness cannot be sustained with mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness.'" *Id.* (citing *In re Kahn*, 441 F.3d 977, 988, 78 USPQ2d 1329, 1336 (Fed. Cir. 2006) and *KSR*, 550 U.S. at ___ , 82 USPQ2d at 1396).

In supporting a rejection, "[t]he examiner bears the initial burden of factually supporting any *prima facie* conclusion of obviousness." *Id.* "If the examiner does not produce a prima facie case, the applicant is under no obligation to submit evidence of nonobviousness." *Id.* (emphasis added).

Finally, "[w]hen determining whether a claim is obvious, an examiner must make "a searching comparison of the claimed invention - *including all its limitations* - with the teaching of the prior art." *In re Wada and Murphy*, Appeal 2007-3733 (BPAI 2008)(citing

In re Ochiai, 71 F.3d 1565, 1572 (Fed. Cir. 1995)). "Thus, '<u>obviousness requires a suggestion of all limitations in a claim</u>.'" *Id* (citing *CFMT, Inc. v. Yieldup Intern. Corp.*, 349 F.3d 1333, 1342 (Fed. Cir. 2003)(citing *In re Royka*, 490 F.2d 981, 985 (CCPA 1974))(emphasis added).

Independent claim 1 recites, inter alia,

> determining whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of a memory [.]

Independent claim 18, which has its own scope, recites a similar feature.

Appellants/Applicants respectfully submit that neither *Bonola* nor the *conventional memory allocation process* have been shown to disclose at least the above feature. Specifically, the primary citation to *Bonola* has not been shown to disclose the above feature. The *conventional memory allocation process* has not been shown to make up for this deficiency. Accordingly, without conceding its propriety, the proposed combination of *Bonola* and the *conventional memory allocation process* is likewise deficient, even in view of the knowledge of one of ordinary skill in the art.

The Final Office Action contends that paragraph 37 of *Bonola* discloses the above feature. Specifically, the Final Office

Action contends *Bonola* "… executes HeapAlloc… examine heap structure [to] determine if there is an appropriately sized heap sub-region in the free list wherein the free list contains unassigned heap sub-regions…". This contention is respectfully traversed.

According to the paragraph of *Bonola* cited (paragraph 37), the N bits of the heap data structure can be used to determine whether there exists an "appropriately sized subregion 302A-302F in [a] free list…". While the technique discussed in *Bonola* may indicate the existence of an unassigned aggregate sub-region containing memory resources of a certain size, it does not indicate whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of memory. It is respectfully submitted that *Bonola* is silent with respect to pre-allocation of sections of a memory with a group size.

With regard to this feature, Appellants/Applicants' specification clearly indicates what is meant by pre-allocation of sections of memory with a group size:

> The memory may be divided into sections, each of which is allocated a size, based on a set of data anticipated to be received by the hardware, before actually receiving any data in the hardware.

Appellants/Applicants' specification, page 4, lines 17-20. This disclosure is in contrast with the disclosure of *Bonola*,

in which unassigned subregions of memory having different sizes are "created" simply by virtue of the assignment of surrounding memory sections (*see Bonola*, paragraph 33). Thus, such subregions are not planned in advance, and therefore cannot be pre-allocated with a size at all, much less pre-allocated with a size based on a set of data anticipated to be received by the hardware.

In view of the above, Appellants/Applicants respectfully submit that *Bonola* cannot properly be relied upon for disclosing the above feature. Further, it is again noted that the above feature is an expressly recited feature in the claims. Accordingly, favorable review and reversal of the rejection under 35 U.S.C. § 103 are respectfully requested.

A PRIMA FACIE CASE OF OBVIOUSNESS OF CLAIMS 1-34 HAS NOT BEEN ESTABLISHED AS IT HAS NOT BEEN SHOWN THAT *BONOLA* DISCLOSES A DYNAMICALLY ALLOCATED SIZE BEING THE SMALLEST-SIZED GROUP ENTRY NECESSARY TO STORE THE PORTION OF THE SET OF DATA

Appellants/Applicants respectfully submit that the record fails to establish that each feature of independent claims 1 and 18 is disclosed by the proposed combination of *Bonola* and the *conventional memory allocation process.* Accordingly, Appellants/Applicants respectfully submit that the record fails to

establish a prima facie case of obviousness.

Independent claim 1 recites, inter alia,

> if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data.

Independent claim 18, which has its own scope, recites a similar feature.

Applicants/Applicants respectfully submit that neither *Bonola* nor the *conventional memory allocation process* have been shown to disclose at least the above feature. Specifically, the primary citation to *Bonola* has not been shown to disclose the above feature. The *conventional memory allocation process* has not been shown to make up for this deficiency. Accordingly, without conceding its propriety, the proposed combination of *Bonola* and the *conventional memory allocation process* is likewise deficient, even in view of the knowledge of one of ordinary skill in the art.

The Final Office Action contends that paragraphs 38-39 of *Bonola* disclose the above feature. Specifically, the Final Office

Action contends "for the purposes of the invention of Bonola, the smallest size needed is equivalent to a heap sub region that is less than twice the memory required for the application…" This contention is respectfully traversed.

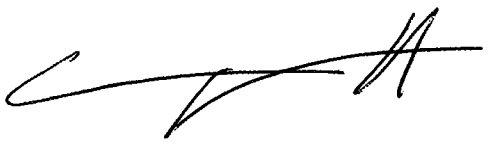
Bonola discusses that after a heap subregion is divided into two heap subregions, if a remaining half of the heap subregion contains twice as much memory as is required by the requesting application, the remaining half of the heap subregion is iteratively divided in half until there is a heap subregion that does not have twice as much memory as is required. Then, the subregion is assigned to the application. Thus, *Bonola* merely reduces the size of the subregion to a threshold size and not the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data.

In view of the above, Appellants/Applicants respectfully submit that *Bonola* cannot properly be relied upon for disclosing the above feature. Further, it is again noted that the above feature is an expressly recited feature in the claims. Accordingly, favorable review and reversal of the rejection under 35 U.S.C. § 103 are respectfully requested.

CONCLUSION

In summary, Appellants/Applicants respectfully submit that claims 1-34 patentably define over the prior art. Reversal of the Examiner's rejections is respectfully requested.

Respectfully Submitted,



Dated: April 13, 2009

Christopher Paul Mitchell
Registration No. 54,946

Dugan & Dugan, PC
Hawthorne, New York
Attorneys for Applicants
(914) 579-2200

CLAIM APPENDIX

Claim 1 (Previously Presented): A method comprising:

receiving a set of data;

determining whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of a memory;

if a free group entry of the size required by the portion of the set of data does not exist in one of the plurality of sections of the memory, determining whether the memory includes one or more sections of an unallocated size; and

if the memory includes one or more sections of an unallocated size, allocating one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the section of the dynamically allocated size including one or more group entries of the size required by the portion of the set of data, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data.

Claim 2 (Previously Presented): The method of claim 1 wherein determining whether a free group entry of the size required by the portion of the set of data exists in the memory pre-allocated with a group size in one of a plurality of sections of the memory includes determining whether a free group entry of the size required by the portion of the set of data for uniquely identifying each portion of the set of data exists in one of the plurality of sections of memory pre-allocated with a group size.

Claim 3 (Original): The method of claim 1 wherein determining whether the memory includes one or more sections of an unallocated size includes accessing a control structure for one or more sections of the memory, the control structure storing information about the structure of a section.

Claim 4 (Original): The method of claim 1 further comprising, from the section of a dynamically allocated size, allocating an initial group entry of the size required by the portion of the set of data for storing the portion of the set of data.

Claim 5 (Original): The method of claim 4 further comprising:
receiving a modified set of data;
determining whether a portion of the modified set of data may be stored more efficiently in a group entry of a different size from another section of the memory such that the aggregate number of unused entries in the group entries used for storing the modified set of data is minimized;
allocating a group entry of the different size required by the portion of the modified set of data from another section of the memory to store the portion of the modified set of data; and
deallocating the initial group entry to the section of memory from which the initial group entry was allocated.

Claim 6 (Original): The method of claim 5 further comprising updating the control structure that stores information about the structure of the other section.

Claim 7 (Original): The method of claim 5 further comprising

updating the control structure that stores information about the structure of the section of memory from which the initial group entry was allocated.

Claim 8 (Original): The method of claim 5 wherein deallocating the initial group entry to the section of memory from which the initial group entry was allocated leaves all entries of the section unused.

Claim 9 (Original): The method of claim 8 further comprising clearing the group entry size allocation of the section.

Claim 10 (Original): The method of claim 1 further comprising, if the memory does not include one or more sections of an unallocated size, determining whether a free group entry of a size larger than the size required by the portion of the data exists, wherein sections allocated to the smallest available size larger than the size required by the portion of the data are checked prior to sections allocated to larger available sizes.

Claim 11 (Original): The method of claim 10 further comprising, if a free group entry of a size larger than the size required by the portion of the data exists in a section allocated to a size larger than the size required by the portion of data, allocating an initial group entry of the size larger than the size required by the portion of the set of data from the section allocated to a size larger than the size required by the portion of the data for storing the portion of the set of data.

Claim 12 (Original): The method of claim 10 further comprising,

if a free group entry of a size larger than the size required by the portion of the data does not exist outputting an error condition.

Claim 13 (Original): The method of claim 11 further comprising:
receiving a modified set of data;
determining whether a portion of the modified set of data may be stored more efficiently in an group entry of a different size from another section of the memory such that the aggregate number of unused entries in the group entries used for storing the modified set of data is minimized;
allocating a group entry of the different size required by the portion of the modified set of data from another section of the memory to store the portion of the modified set of data; and
deallocating the initial group entry to the section of memory from which the initial group entry was allocated.

Claim 14 (Original): The method of claim 13 further comprising updating the control structure that stores information about the structure of the other section.

Claim 15 (Original): The method of claim 13 further comprising updating the control structure that stores information about the structure of the section of memory from which the initial group entry was allocated.

Claim 16 (Original): The method of claim 13 wherein deallocating the initial group entry to the section of memory from which the initial group entry was allocated leaves all entries of the section unused.

Claim 17 (Original): The method of claim 16 further comprising clearing the group entry size allocation of the section.

Claim 18 (Previously Presented): An apparatus comprising:

a memory;

a plurality of registers; and

dynamic allocation logic coupled to the memory and the plurality of registers, and adapted to:

receive a set of data;

determine whether a free group entry of a size required by a portion of the set of data exists in memory pre-allocated with a group size in one of a plurality of sections of the memory;

if a free group entry of the size required by the portion of the set of data does not exist in one of the plurality of sections of the memory, determine whether the memory includes one or more sections of an unallocated size; and

if the memory includes one or more sections of an unallocated size, allocate one of the sections of an unallocated size to the size required by the portion of the set of data thereby creating a section of a dynamically allocated size, the dynamically allocated size being the smallest-sized group entry necessary to store the portion of the set of data.

Claim 19 (Previously Presented): The apparatus of claim 18 wherein the dynamic allocation logic is further adapted to determine whether a free group entry of the size required by the portion of the set of data for uniquely identifying each portion of the set of data exists in one of the plurality of sections of memory pre-allocated with a group size.

Claim 20 (Original): The apparatus of claim 18 wherein the dynamic allocation logic is further adapted to access a control structure for one or more sections of the memory, the control structure storing information about the structure of a section.

Claim 21 (Original): The apparatus of claim 18 wherein the dynamic allocation logic is further adapted to, from the section of a dynamically allocated size, allocate an initial group entry of the size required by the portion of the set of data for storing the portion of the set of data.

Claim 22 (Original): The apparatus of claim 21 wherein the dynamic allocation logic is further adapted to:

receive a modified set of data;

determine whether a portion of the modified set of data may be stored more efficiently in a group entry of a different size from another section of the memory such that the aggregate number of unused entries in the group entries used for storing the modified set of data is minimized;

allocate a group entry of the different size required by the portion of the modified set of data from another section of the memory to store the portion of the modified set of data; and

deallocate the initial group entry to the section of memory from which the initial group entry was allocated.

Claim 23 (Original): The apparatus of claim 22 wherein the dynamic allocation logic is further adapted to update the control structure that stores information about the structure of the other

section.

Claim 24 (Original): The apparatus of claim 22 wherein the dynamic allocation logic is further adapted to update the control structure that stores information about the structure of the section of memory from which the initial group entry was allocated.

Claim 25 (Original): The apparatus of claim 22 wherein the dynamic allocation logic is further adapted to deallocate the initial group entry to the section of memory from which the initial group entry was allocated leaving all entries of the section unused.

Claim 26 (Original): The apparatus of claim 25 wherein the dynamic allocation logic is further adapted to clear the group entry size allocation of the section.

Claim 27 (Original): The apparatus of claim 18 wherein the dynamic allocation logic is further adapted to, if the memory does not include one or more sections of an unallocated size, determine whether a free group entry of a size larger than the size required by the portion of the data exists, wherein sections allocated to the smallest available size larger than the size required by the portion of the data are checked prior to sections allocated to larger available sizes.

Claim 28 (Original): The apparatus of claim 27 wherein the dynamic allocation logic is further adapted to, if a free entry group of a size larger than the size required by the portion of the data exists in a section allocated to a size larger than the size required

by the portion of data, allocate an initial group entry of the size larger than the size required by the portion of the set of data from the section allocated to a size larger than the size required by the portion of the data for storing the portion of the set of data.

Claim 29 (Original): The apparatus of claim 27 wherein the dynamic allocation logic is further adapted to, if a free group entry of a size larger than the size required by the portion of the data does not exist, output an error condition.

Claim 30 (Original): The apparatus of claim 23 wherein the dynamic allocation logic is further adapted to:

receive a modified set of data;

determine whether a portion of the modified set of data may be stored more efficiently in a group entry of a different size from another section of the memory such that the aggregate number of unused entries in the group entries used for storing the modified set of data is minimized;

allocate a group entry of the different size required by the portion of the modified set of data from another section of the memory to store the portion of the modified set of data; and

deallocate the initial group entry to the section of memory from which the initial group entry was allocated.

Claim 31 (Original): The apparatus of claim 30 wherein the dynamic allocation logic is further adapted to update the control structure that stores information about the structure of the other section.

Claim 32 (Original): The apparatus of claim 30 wherein the dynamic allocation logic is further adapted to update the control structure that stores information about the structure of the section of memory from which the initial group entry was allocated.

Claim 33 (Original): The apparatus of claim 30 wherein the dynamic allocation logic is further adapted to deallocate the initial group entry to the section of memory from which the initial group entry was allocated leaving all entries of the section unused.

Claim 34 (Original): The apparatus of claim 33 wherein the dynamic allocation logic is further adapted to clear the group entry size allocation of the section.

EVIDENCE APPENDIX

Not applicable

RELATED PROCEEDINGS APPENDIX

Not applicable